IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICANT:	Richard W. Pratt
SERIAL NO.:	08/924,785
FILING DATE:	September 5, 1997
TITLE:	SYSTEM AND METHOD FOR REMOTE DEVICE MANAGEMENT
EXAMINER:	Prieto, Beatriz
ART UNIT:	2142

APPEAL BRIEF

COMMISSIONER FOR PATENTS
P.O. BOX 1450
ALEXANDRIA, VA 22313-1450

Dear Sir:

This paper is in support of a Notice to Appeal from the Office Action dated April 21, 2004 to the Board of Patent Appeals and Interferences. Please consider the following.

REMARKS

Real Party in Interest

The real party in interest is Cisco Technology, Inc. as assignee of the present patent application. (See Reel/Frame: 9159/0429 and 9458/0130.)

Related Appeals and Interferences

There are no other appeals or interferences known which will directly affect or be directly affected by or have a bearing on the present appeal.

Status of Claims

The total range of claims prosecuted in this application since the beginning is from 1 to 74. Claims 1-43, 45-61, 65, and 66 have been canceled. Claims 44, 62-64, and 67-74 stand rejected and are appealed.

Status of Amendments

An Amendment filed on a date even herewith is being presented in partial response to the Office Action dated April 21, 2004. It has not yet been acted upon or entered.

The Amendment merely proposes to cancel two claims and thereby narrow the issues for this appeal. Entry of the Amendment is appropriate and is respectfully requested.

Summary of Invention

"In accordance with the present invention, a system for enabling remote management of a network device such as a web server, a router, or any networked computer is provided." (Page 3, lines 2-4) As illustrated in FIG. 1, "[t]he system 100 includes a remote client 110, a programmer

client 142, and a network device 120, each coupled together via a communications channel 165. The communications channel 165 can be a direct connection through a direct connect cable, an Internet connection, or any other means of intercommunication." (Page 8, lines 2-6) "A [network device control] software program controls the network device to perform an intended function (such as routing IP packets). To enable configuration and management of the software program, device-specific downloadable units are embedded in the binary file of the software program. A downloadable unit is any unit of code that can be downloaded to a web engine for performing a particular function." (Page 3, lines 5-10) "It will be appreciated [by one of ordinary skill in the art] that the downloadable unit code 320 [of FIG. 3] is embedded in the binary file 300 [of FIG. 3] so that the downloadable units 190 [of FIG. 1] can be extracted from the software program 180 [of FIG. 1] while the software program 180 is executing." (Page 12, lines 19-22) "Each embedded downloadable unit may contain a communicator component, an interface component, and a configuration component." (Page 3, lines 14-15) "One skilled in the art will [further] appreciate that the remote client may retrieve the downloadable unit each time network device management is desired, thereby enabling the programmer [client] to update available features, available control options, or version information by modifying only the downloadable unit." (Page 3, line 20 through page 4, line 1)

"Enabling [the network manager user of] a remote client to download a software management mechanism directly from the software program enables the network manager to have the most up-to-date network device management software and the most up-to-date functions. This also insures that the network device management software is compatible with the version of the software running on the remote [client] device. Users need not maintain configuration files for each network device on their local data storage. Further, to update or

improve the available functions, a revised downloadable unit may be embedded in the binary file of the software program [by the programmer client]. The user need not independently obtain the new network device management software since it is readily available from the network device. The user need not independently obtain new network device management software each time the software program running on the device is updated [by the programmer client]." (Page 6, lines 3-13)

According to exemplary claim 44, what is claimed is a method for modifying available remote device management services of a network device. (See 142 of FIG. 1 and page 9, lines 3-12, FIG. 6 and page 16, lines 1-19, and FIG. 8 and page 18, lines 7-19, among others.) The method comprises four basic elements. The first basic element includes obtaining a new downloadable unit for performing a new service. (See 190 of FIG. 1, 320 of FIG. 3, and FIG 4.) The new downloadable unit includes three sub-elements. The first sub-element is a communicator component for establishing a communications channel between a remote client and the network device. (See 410 of FIG. 4 and 165, 110, and 120 of FIG. 1.) The second sub-element is an interface component for enabling the remote client to communicate with the new downloadable unit. (See 420 of FIG. 4.) The third sub-element is a configuration component for managing the network device. (See 430 of FIG. 4.) The second basic element includes retrieving a network device control software program binary file having an embedded old downloadable unit for performing an old service from the network device. (See 142, 120, 180, and 190 of FIG. 1, 300 and 320 of FIG. 3, and FIG. 8.) The third basic element includes substituting the old downloadable unit with the new downloadable unit. (See 142 and 148 of FIG. 1 and 610 and 640 of FIG. 6.) The fourth basic element includes loading the network device control software program binary file having the new downloadable unit embedded in the

binary file onto the network device. (See 142 and 144 of FIG. 1 and 660 and 670 of FIG. 6.) The claim concludes by noting that the network device control software program is executed by the network device. (See page 12, lines 19-22.)

Issues

Whether claims 44, 62-64, and 67-74 are unpatentable under 35 U.S.C. § 103(a) over *Madany* (US 5,922,050) in view of *Beard* (US 6,067,577) in further view of *Nakagawa et al.* (US 5,832,911).

Grouping of Claims

The claims 44, 62-64, and 67-74 stand or fall together.

Argument

Claims 44 and 62-74 stand rejected under 35 U.S.C. § 103(a) as being allegedly unpatentable over *Madany* (US 5,922,050) in view of *Beard* (US 6,067,577) in further view of *Nakagawa et al.* (US 5,832,911). This rejection is respectfully traversed.

Generally with reference to claim 44 as outlined in the Summary of Invention above, the Office Action states that the first basic element and its three sub-elements are disclosed or suggested by *Madany*, the second basic element is disclosed or suggested by *Beard*, and that the third and fourth basic elements are disclosed or suggested by *Nakagawa*. However, the following will demonstrate that the features disclosed in *Beard* and *Nakagawa* may not be properly combined with the features disclosed in *Madany*.

Briefly, the Office Action overextends the primary reference of *Madany* and uses this misinterpretation as the basis for combining with the other two references. Properly understood, the references can not be combined. Further, the Office Action fails to provide proper motivation for the proposed combination. Generic recitations of alleged elements or limitations by *Beard* and *Nakagawa* do not provide sufficient basis for combining these elements or limitations with *Madany*. Consequently, the prior art fails to disclose or suggest "retrieving a network device control software program binary file having an embedded old downloadable unit for performing an old service from the network device", "substituting the old downloadable unit with the new downloadable unit", and "loading the network device control software program binary file having the new downloadable unit embedded in the binary file onto the network device" as claimed.

According to M.P.E.P. § 2141.02, "[I]n determining the differences between the prior art and the claims, the question under 35 U.S.C. § 103 is not whether the differences *themselves* would have been obvious, but whether the claimed invention *as a whole* would have been obvious." The Office Action employs loose citations and logic to result in a rejection that is in violation of this principle. Two individual motivations for prior art differences are provided with no single motivation for the invention as a whole as required. Even if the individual motivations are found to be acceptable in light of the discussion to follow, the two individual motivations provided do not motivate the claimed invention as a whole. This alone is independent grounds for withdrawing the rejection. There must be a showing that the prior art had a specific understanding and motivation to make the claimed combination and not just each individual modification. The Applicant respectfully asserts that such a showing has not been made and can not be made.

Turning briefly to the individual motivations, the Office Action first admits that *Madany* does not specifically teach the second basic element, then alleges that *Beard* does and that combining *Madany* with *Beard* would have been obvious. The motivation essentially being that it is possible to do so. Second, the Office Action admits that neither *Madany* nor *Beard* specifically teaches the third and fourth basic elements, then alleges that *Nakagawa* does and that combining *Madany* and *Beard* with *Nakagawa* would have been obvious. The motivation asserted in the Office Action is essentially that it would be beneficial to do so in any context. These individual motivations will be shown to be inadequate below. If either of the individual motivations are found to be insufficient, then this provides further independent grounds for withdrawing the rejection.

The core of the rejection is the system taught by *Madany*. Consequently, a detailed review of *Madany* is in order. The context and boundaries of the teachings of *Madany* are important to consider when determining what one of ordinary skill in the art would garner from the reference. According to the Field of Invention section of *Madany*, "[t]he present invention provides a system for permitting low-cost devices to be controlled by programs contained in the device." (Col. 1, lines 7-9)

The low-cost theme is repeated consistently in the Background section of *Madany* as evidenced by the following citations from column 1, line 11 through column 2, line 7 (emphasis added below).

> Various devices are available [in the prior art] which include embedded control programs and processing hardware to execute the programs. These devices may contain small computers having a processor and memory or storage for executing the programs stored

within the device. Typically, this computing hardware is provided in moderately or higher-priced devices. It is *not commercially feasible* to provide embedded computer hardware which is several times greater than the *typical selling price of the device*. For example, embedding $50.00 of computer hardware in a $10.00 device is not likely to be commercially feasible.

Although low-performance computer hardware may be embedded in a device at a low cost, this *low-performance hardware* may not be capable of executing programs written in programming languages requiring significant processing capabilities. Programs written in high level programming languages, such as object-oriented languages, may not be capable of execution by inexpensive computer hardware which lacks significant processing ability.

Various devices [according to the present invention] may be interconnected with other devices or computers across a network or other communication medium. These devices include *low-cost items* such as light switches and door locks as well as *more expensive items* such as televisions and VCRs. By interconnecting various devices, a particular device may be controlled manually by a user, or may be controlled automatically by another device or computer coupled to the communication medium. To accomplish this control, the low-cost device must contain a sufficient amount of processing hardware to communicate across the communication medium and control operation of the device itself. However, this *level of hardware may not be sufficient* to execute programs written in high level programming languages which may be necessary to control operation of the device. As discussed above, the addition of hardware sufficient to execute such programs may *increase the cost* of the device to a point where the device is *not competitive*.

A specific programming language capable of controlling consumer devices is the JAVA™ programming language environment developed by Sun Microsystems...JAVA is an object-oriented and architecture-neutral programming language environment...

JAVA programs may be executed on any computer platform that supports a JAVA virtual machine (the JAVA virtual machine executes JAVA byte-codes). However, as with other high level languages, the minimum computing resources and memory requirements for a device capable of executing a JAVA program would *increase the price of a low-cost device beyond commercial feasibility*.

Therefore, a mechanism is necessary [but not available in the prior art] to permit a *low-cost device* to be controlled by an application program requiring a significant amount of computing hardware, yet maintaining a relatively *low cost* for the device.

In large part, the low cost emphasis is generally achieved in the system of *Madany* through reducing the processing hardware and software to the bare minimum in the devices and concentrating the cost in one or more computers. This is evidenced in the Summary of Invention of *Madany* as follows:

> Embodiments of the present invention provide a system for permitting low-cost devices to be controlled by programs contained within the device. Although the low-cost device contains processing hardware and a program or applet for controlling the device, the device's processing hardware is incapable of executing the program. The invention transmits the program stored on the device across a network or communication medium to a computer capable of executing the program. The computer executes the program and generates control signals for controlling the device. The invention maintains the low cost of the device while permitting control of the device by a program incapable of execution by the device itself. (Col. 2, lines 10-23)

The underlying assumption appears to be that the consumer already has or is willing to buy at least one high cost device to realize the system described by *Madany*. More than one high cost device is not required. Since most consumers already have at least one general purpose computer, this assumption is easily validated.

Against this background of *Madany*, the Office Action implicitly disregards the low cost theme based apparently on one word in the disclosure by *Madany*. This one word is used to justify combining references on opposite ends of the cost and processing power spectrum. *Madany* is on one end and *Beard* and *Nakagawa* are on the other. The Applicant respectfully contends that any combination of these references is illegitimate.

Turning now to FIGS. 1 and 7, *Madany* discloses a network including two computers and multiple devices. The devices are described further with respect to FIGS. 2, 4, and 5. The computers are described further with respect to FIGS. 3 and 6. Predominantly, the devices and computers are described independently of one another. The computer/device dichotomy is important to *Madany* as this is how he keeps the overall system cost down as noted above. The computers do substantially all of the processing while the devices do effectively none.

The computer/device dichotomy is broken in one peculiar instance. The one and only exception to the dichotomy is on column 3, line 45 where *Madany* lists a "computer" as being a possible example device. The inclusion of a "computer" as a computer-type device in the list appears to be unusual as it is out of place with the other examples. The other examples include light switches, televisions, radios, door locks, telephones, coffee makers, security systems, and VCRs. One of ordinary skill would recognize that the other example devices are dedicated devices and not general purpose devices like the computer-type device might be. Generally one would recognize *Madany* as a home automation system. To further emphasize the unusualness of including the computer-type device in the list of examples is the fact that there is no further special discussion by *Madany* of how a "computer" would operate as a device (16, 18, 20 of FIG. 1) and not as a computer (10, 12 of FIG. 1) or as both a device and a computer in his system. One of ordinary skill would reason that the general purpose nature of a computer-type device might require special attention. For example, there is no explanation as to why a computer-type device that might be able to perform its own processing would have another computer do the processing for it or have a second *Madany* device processor added to it. This does not fit well into the low cost model of *Madany*. No special attention is given to the computer-type device by *Madany*. The disclosure is essentially one word long. Consequently, one of ordinary skill in the art would discount the recitation of a possible computer-type device by *Madany* as unsupported or erroneous. The Office Action should do likewise and withdraw the rejection.

Even if not fully discounted, the Office Action can not necessarily assume implicitly that the computer-type device operates any differently than the other devices listed and described as devices. The use by *Madany* of the term computer alone is not grounds to make broad assumptions about what hardware or software a computer-type device might include beyond that

which is explicitly described by *Madany* with respect to FIG. 2. If it is not taught by *Madany*, then any adaptations or assumptions must be properly cited and motivated to avoid over-extending the reference. A detailed discussion of the disclosure of *Madany* with respect to FIG. 2 follows. Open speculation by the Office Action is not enabled or allowed.

The primary discussion of the embodiment of the device, whatever other name the device might go by, is given by *Madany* with respect to FIG. 2 on column 3, line 41 through column 4, line 42. For discussion purposes, the details shown in FIG. 2 of *Madany* will be referred to here as network control logic. Such logic may include hardware, software, or both. One will realize that the example devices disclosed by *Madany* already exist and that he is adding the network control logic to the device so that it can perform a network control function. The existing device already has all of the original dedicated logic, if any, necessary to perform its respective original dedicated functions such as turning on lights, brewing coffee, and playing video tapes. Nevertheless, *Madany* does not disclose how his network control logic is to be integrated into the existing device. He describes the network control logic in the abstract only. For example, there is no discussion of specialized hardware or software, such as speakers, microphones, buttons, switches, valves, motors, and the like, that might be needed to perform the original dedicated functions of the device. Further, there is no discussion of the network control logic replacing the existing original dedicated logic or vice versa, that is, the network control logic must be in addition to and not in place of the original dedicated logic. Similarly, there is no discussion of the network control logic performing any of the original dedicated functions or vice versa. In fact, the only network control functions required by *Madany* are essentially that the device be able to (1) store/retrieve an applet, (2) communicate over the network, and (3) respond to control signals from the computer running the applet. (See col. 3, lines 53-56.) The device is designed

to still respond to control signals directly from the user. As presented by *Madany*, the network control logic is very limited to keep costs down, that is, the addition of the network control logic is not supposed to add substantially to the cost of the device. This is especially true, for example, when one is considering the cost of a light switch. There is no discussion by *Madany* of economies or synergies achieved by combining or substituting logic or functions. Some of the devices have no original dedicated software. Consequently, one can not necessarily assume implicitly that combinations or substitutions of logic or functions would be possible or supported. Further, except for some device specific special implementation details, any proposed combinations or substitutions should work equally well for any of the listed devices as *Madany* never distinguishes one from the other. Thus, the combinations or substitutions proposed by the Office Action either explicitly or implicitly should work equally well, for example, for the light switch and the VCR. Again, if it is not taught by *Madany*, then any and all adaptations or assumptions by the Office Action must be properly cited and motivated to avoid over-extending the teachings. Open speculation is not enabled or allowed.

A careful review of *Madany* will demonstrate how limited the network control logic is disclosed to be. Of course it is recognized that the discussion is presented by *Madany* as being a minimum standard. However, anything proposed by the Office Action beyond the minimum has to be justified against the context in which *Madany* presents his system and can not just be implicitly assumed. *Madany* starts his discussion as follows:

> The device shown in FIG. 2 includes a processor 22, a read only memory (ROM) 24, and a random access memory (RAM) 26. Processor 22 may be an inexpensive processor capable of performing basic control functions and communication functions across the network [14 of FIG. 1]. Similarly, ROM 24 and RAM 26 may be relatively small to reduce the overall cost of the device. (Col. 3, lines 46-52)

Madany later reveals the following:

> ROM 24 also includes a program or applet containing information regarding the features of the device and instructions for providing control signals to the device. The applet contains all of the information necessary to describe, control, and communicate with the device. This applet cannot be executed by the processor 22 due to the inadequate processing resources of the processor. (Col. 4, lines 23-30)

Taken together, these quotations indicate that the device processor 22 of *Madany* is inexpensive and low power. This is emphasized in the independent claims of *Madany* where the device is claimed as being "unable to execute said program [or applet] code". This description by *Madany* excludes a whole range of possible hardware that might be erroneously equated with the processor 22 based on FIG. 2 alone. As noted above the device only performs three basic functions. Whether this is to keep the processor simple or is because the processor is simple is not known. Nevertheless, one supports the other. To change one, one would probably have to change the other and provide proper justification for both.

Similarly taken together, these two quotations indicate that the ROM 24 is low cost and includes the applet. There is no discussion of altering or replacing the ROM 24 by *Madany*. The applet is static in the device. The applet is not running nor is it embedded in anything that is running. There is no discussion of multiple applets on a device whether the applets are new or old. Double citations to *Madany* by the Office Action does not turn one applet into two to support the rejection. There is no discussion of updating the applet by *Madany*. In fact, there is little motivation to do so. These and other deficiencies in *Madany* are enough grounds for withdrawal of the rejection. However, any and all deficiencies are alleged by the Office Action to be disclosed or suggested by the other two references which will be considered in turn next.

The Office Action alleges that *Beard* discloses or suggests "retrieving" "embedded" software as claimed and that it would have been obvious to combine the references. This argument has a number of defects. The argument fails to note where the applet of *Madany* would supposedly be embedded. *Madany* does not have access to the original dedicated logic of the device, if any, for embedding into thus closing off this option. This leaves the network control logic. However, as noted above, the network control logic is limited and embedding into it may not even be possible or desirable. Further, the argument fails to note that the device lacks the ability to perform the function of extracting the applet from wherever it might be embedded. Recall that there are only three network control functions that the device performs and extracting is not one of them. Adding functions increases the device cost and complexity. Further still, the argument fails to motivate embedding over storing. *Madany* chooses to store the applet in the ROM 24. The Office Action states that embedding is an option that would work as well. The suggestion being that choosing one over the other would be a simple design choice. However, embedding would be more expensive than storing and, as evidenced above, *Madany* does everything in his power to avoid increasing the cost of the device. Thus, *Madany* teaches away from the embedding choice even if it were possible. Consequently, the features disclosed in *Beard* may not be properly combined with the features disclosed in *Madany*.

The Office Action goes on to allege that *Nakagawa* discloses or suggests "substituting" and "loading" software over a network as claimed and that it would have been obvious to combine the references. This argument is also defective. The argument fails to take *Madany* in context. The system disclosed by *Nakagawa* is directed to general purpose computers. (See FIG. 1, among others.) By contrast, *Madany* is directed to dedicated devices. While one might change software on a general purpose device, one is not likely to change software on a low-cost

dedicated device. One is more likely just to purchase a new or additional dedicated device as the software tends to constitute most of the cost of the device for low-cost devices such as those listed by *Madany*. Vague notions in the Office Action of enhancing software capabilities by freely updating it must be supported by a specific need and ability to perform such updates. The cited references and Office Action fail to disclose such a need in the context of *Madany*. Citations by the Office Action to *Nakagawa* are therefore misplaced. Further, software updates are not possible in the device disclosed by *Madany*. The applet is stored in the ROM 24 which is static. There is no disclosure of swapping out ROMs or using other forms of non-static memory for storing the applet. Further still, software changes or the enablement of software changes would entail additional costs to the device which *Madany* is attempting to keep to a minimum. Thus, *Madany* teaches away from the software changes even if it were possible. Consequently, the features disclosed in *Nakagawa* may not be properly combined with the features disclosed in *Madany* and *Beard*.

The presently claimed invention directly or indirectly addresses one or both of at least two problems never recognized or addressed by the cited prior art. For example, the specification notes that one problem relates to interface access capabilities from a remote site. "If a network manager needs to manage a network device on the network from the remote site, a proprietary interface site component including supporting software must be installed at the remote site. This limits the locations from which management can occur and reduces the usefulness of the proprietary interface." (Page 1, line 22 through page 2, line 2) The specification goes on to note that a second problem relates to the compatibility between the proprietary interface and the network device.

> For example, routers run an operating system such as the Internetwork Operating System (IOS) produced by Cisco [Systems], Inc. The IOS controls routing by finding optimal paths and redirecting traffic quickly around network failures. A network manager who wants to access the router to monitor such a failure must have software compatible with the version of the IOS running on the device. Additionally, each device may perform different tasks requiring the use of device-specific programs. Storing device-specific configuration and program file would not be a problem for a single network device. However, it is not feasible to maintain these files for the myriad of devices in most networks. (Page 2, lines 4-12)

In view of these problems, the specification concludes that "...a system and method for using device-specific programs and configuration data to manage a network device from a remote site are needed." (Page 2, lines 12-14)

Given the above discussion and arguments, one can not support the contention that the present claims are rendered obvious by the cited references. Consequently, it is respectfully requested that the appeal be granted and that the rejection be withdrawn.

Request for Allowance

In view of the foregoing, reconsideration and an early allowance of this application are earnestly solicited.

If any matters remain which could be resolved in a telephone interview between the Examiner and the undersigned, the Examiner is invited to call the undersigned to expedite resolution of any such matters. Please charge any additional required fee or credit any overpayment not otherwise paid or credited to our deposit account No. 50-1698.

Respectfully submitted,
THELEN, REID, & PRIEST LLP

Dated: July 12, 2004

David B. Ritchie
Reg. No. 31,562

Thelen, Reid, & Priest LLP
P.O. Box 640640
San Jose, CA 95164-0640
Tel. (408) 292-5800
Fax (408) 287-8040

APPENDIX:

The following listing of claims contains a copy of the claims involved in the Appeal:

1-43. (Canceled)

44. (Previously Presented) A method for modifying available remote device management services of a network device, the method comprising:

obtaining a new downloadable unit for performing a new service, the new downloadable unit including:

a communicator component for establishing a communications channel between a remote client and the network device,

an interface component for enabling the remote client to communicate with the new downloadable unit, and

a configuration component for managing the network device;

retrieving a network device control software program binary file having an embedded old downloadable unit for performing an old service from the network device;

substituting the old downloadable unit with the new downloadable unit; and

loading the network device control software program binary file having the new downloadable unit embedded in the binary file onto the network device,

wherein the network device control software program is executed by the network device.

45-61. (Canceled)

62. (Previously Presented) A computer-readable medium having stored thereon computer-executable instructions for performing a method of modifying available remote device management services of a network device, the method comprising:

obtaining a new downloadable unit for performing a new service, the new downloadable unit including:

a communicator component for establishing a communications channel between a remote client and the network device,

an interface component for enabling the remote client to communicate with the new downloadable unit, and

a configuration component for managing the network device;

retrieving a network device control software program binary file having an embedded old downloadable unit for performing an old service from the network device;

substituting the old downloadable unit with the new downloadable unit; and

loading the network device control software program binary file having the new downloadable unit embedded in the binary file onto the network device,

wherein the network device control software program is executed by the network device.

63. (Previously Presented) An apparatus for modifying available remote device management services of a network device, the apparatus comprising:

means for obtaining a new downloadable unit for performing a new service, the new downloadable unit including:

a communicator component for establishing a communications channel between a remote client and the network device,

an interface component for enabling the remote client to communicate with the new downloadable unit, and

a configuration component for managing the network device;

means for retrieving a network device control software program binary file having an embedded old downloadable unit for performing an old service from the network device;

means for substituting the old downloadable unit with the new downloadable unit; and

means for loading the network device control software program binary file having the new downloadable unit embedded in the binary file onto the network device,

wherein the network device control software program is executed by the network device.

64. (Previously Presented) The apparatus as defined in claim 63, wherein the network device includes a network router.

65, 66. (Canceled)

67. (Previously Presented) The apparatus as defined in claim 63, wherein the new downloadable unit includes more than one new downloadable unit.

68. (Previously Presented) The apparatus as defined in claim 67, wherein the new downloadable units have been combined into downloadable unit bundles.

69. (Previously Presented) The apparatus as defined in claim 68, wherein the new downloadable units have been combined into downloadable unit bundles according to downloadable unit function.

70. (Previously Presented) The apparatus as defined in claim 68, wherein the new downloadable units have been combined into downloadable unit bundles according to version information.

71. (Previously Presented) The apparatus as defined in claim 63, wherein the network device control software program includes an operating system.

72. (Previously Presented) The apparatus as defined in claim 71, wherein the network device includes a router.

73. (Previously Presented) The apparatus as defined in claim 63, wherein the network device control software program includes a list of available management services.

74. (Previously Presented) The apparatus as defined in claim 73, further comprising a downloadable unit for each of the available management services.